Web 结构挖掘中的 PageRank 算法改进

Improvement of PageRank Algorithm in Web Structure Mining

钱　杰[1]，张　健[2]，高　乐[1]
（1 浙江工业大学 信息工程学院 浙江杭州 310023）
（2 浙江工业大学 软件学院 浙江杭州 310023）

摘　要：本文介绍了 Web 结构挖掘的基本原理，详细分析 Google 的 PageRank 算法，针对其中的缺陷进行改进，提出了一种集链接、时间、网页内容为一体的 PageRank 改进算法 CTPR，目的是将内容与搜索内容相关度高的、比较权威的、新的网页排列在搜索结果的最前面。网页的等级由 CTPR 值决定，CTPR 值由两个部分组成，一个是传统 PR 算法的值；另一个是网页自评值，它与内容权值成正比，与网页的内容的新旧程度成反比。最后，对此算法进行效果演示，验证算法的有效性。

关键词：PageRank　Web 结构挖掘　CTPR 算法　Web 数据挖掘　内容权值　时间权值

1　引言

目前，Web 已成为一个巨大的、分布广泛的和全球性的信息服务中心，它涉及新闻、金融管理、教育、广告、娱乐、电子商务和许多其它信息服务。互联网发展到今天，其规模之大、品种之全是超出人们想象，据权威机构统计，目前网上约有数十亿的网页，信息量成指数级增长。然而，这些知识信息却不能被人们有效的利用，形成一种“Rich Data Poor Information”的状况。目前各种搜索引擎均存在查准率、查全率不高的现象，这种现状无法适应用户对高质量的网络信息服务的需求。

搜索引擎 Google 的成功，是因为它采取了有效的 Web 挖掘技术。Web 信息挖掘就是从包括 Web 页面内容结构、页面链接、用户访问日志等多种 Web 数据源中，抽取感兴趣的潜在的有用模式和隐藏信息的过程。Google 使用的 PageRank 算法就是针对 Web 超链接结构的。

本文将分析 PageRank 算法并提出一种 PageRank 的改进算法。

2　Web 数据挖掘概述

Web 数据挖掘是从数据挖掘发展而来，是数据挖掘技术在 Web 技术中的应用。Web 挖掘综合运用了统计学、计算机网络、数据库与数据仓库、可视化等众多领域的技术。Web 数据挖掘是指从大量、异质、分布的 Web 文档的集合中抽取感兴趣的、有用的模式和隐含信息[1]。一般地，根据人们对数据兴趣的不同，Web 数据分为：内容数据、结构数据、用户使用挖掘。相应地，Web 挖掘也分为三类：Web 内容挖掘（Web Content Mining）、Web 结构挖掘（Web Structure Mining ）和 Web 使用挖掘（Web Usage Mining）。Web 挖掘结构如图 1 所示。

Web 内容挖掘是从文档内容及其描述中抽取知识得过程，可直接对 Web 页面文档内容以及搜索引擎的查询结果进行文本的总结、分类、聚类、关联、分析等。主要包括文本挖掘和多媒体挖掘两类。Web 结构挖掘是指挖掘 Web 潜在链接结构模式，即通过分析页面之间链接、链接和被链接的数量来建立 Web 自身的链接结构模式。Web 结构挖掘可以分为外部结构挖掘、内部结构挖掘以及 URL 挖掘。Web 使用挖掘主要目标是 Web 访问记录中获取有价值的信息，即通过分析 Web 日志数据及相关数据，来发现访问者访问 Web 页面的模式，分析日志记录中的规律。[2]

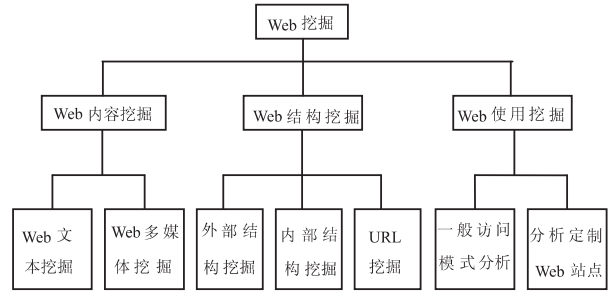

图 1　Web 挖掘结构图

3 Web 结构挖掘概述

3.1 概述

Web 结构挖掘作为 Web 挖掘三大分支之一,越来越受到广大研究机构和科研团体的重视。Web 结构挖掘的对象是 Web 本身的超连接,即通过分析不同网页之间的超链接结构、网页内部结构,以及文档 URL 中的目录路径结构等,发现蕴涵在 Web 文本内容之外的对我们有潜在价值的模式和知识的过程[3]。这种模式可以应用于网页的归类,并且可以由此获取有关不同网页间相似度及关联度的信息,如文档之间的包含、引用或者从属关系。利用这些信息,可以对页面进行排序,发现重要的内容页面,重新组织内容结构,使内容逻辑结构更加合理。有助于用户找到相关主题的权威站点和网页,对网络资源检索结果的排序也有很大的意义。

3.2 Web 结构挖掘算法

目前,Web 用户主要是使用搜索引擎在互联网上检索信息,但目前的搜索引擎返回给用户的往往是成千上万个页面,而且其中很大一部分是重复的或与用户检索要求相关度较低或是不相关的内容。要高效率地利用 Web 提供给我们的资源,必须要充分利用 Web 独有的结构特点,因为 Web 所包含的不仅是页面的内容,同时还有一页面到另一页面的大量的超链接。

为了解决以上问题,一些 Web 结构挖掘的算法应运而生。其中最典型的两个就是 PageRank 算法和 HITS(Hypertext Induced Topic Selection)算法。本文只介绍 PageRank 算法。

3.2.1 PageRank 算法

PageRank 算法由 Stanford 大学的 Brin 和 Page 提出的[4],是 Web 超链接结构分析中最早、最成功的算法之一。PageRank 背后的概念是,每个到页面的链接都是对该页面的一次投票,被链接的越多,就意味着被其他网站投票越多。这个就是所谓的"链接流行度"[5]——衡量多少人愿意将他们的网站和你的网站挂钩。Google 的 PageRank 根据网站的外部链接和内部链接的数量和质量俩衡量网站的价值。就是利用该算法和链接文本标记、词频统计等因素相结合的方法对检索出的大量结果进行相关度排序,将等级值高的网页尽量排在前面。

PageRank 算法的基本思想如下:将整个网络对应成有向图,每个网页是图中的一个节点,每个链接是一条有向边。若 A 指向 B,则说明存在页面 A 指向页面 B 的链接。我们假定页面 A 有 $T_1 \ldots T_n$ 这些页面指向它(即 $T_1 \ldots T_n$ 引用页面 A 或 $T_1 \ldots T_n$ 各自对页面 A 投一票)。参数 d 是一个设置于 0 与 1 之间的阻尼系数,通常设置 d 为 0.85。另外,C(A) 定义为从页面 A 出发的连接数量。则 PageRank 算法的具体迭代公式为:

$$PR(A) = (1-d) + d((RR(T_1)/(C(T_1)) + L + PR(T_n)/C(T_n))$$

说明一下这个公式的意义。

1. $PR(T_n)/C(T_n)$ 的意义为:页面 A 有来自页面 n 的一个反向链接,页面 A 将获得的选票份额是 $PR(T_n)/C(T_n)$;[6]

2. $d(PR(T_1)/C(T_1) + \ldots + PR(T_n)/C(T_n))$ 的意义为:所有这些选票项加在一起;但为了不让其他页面有过大的影响,这个总选票数通过与 0.85(系数 d)相乘而被"阻尼"掉[6];

3. (1 – d)的意义为:(1 – d)这一部分采用概率数学的一个性质,使得"所有页面的 RageRank 值平均数为 1":它弥补了后面那一部分与 d 相乘时失去的那一部分。这同样意味着如果一个页面没有链接指向它(无反向链接),即使如此它仍将有一个很小的 PR 值 0.15(即 1 – 0.85)。

PageRank 算法的实现过程为:将网页的 URL 对应成唯一的整数,把每一个超链接用其整数 ID 存放到索引数据库中,经过预处理之后,设每个网页的初始 PR 值为 1,通过以上的递归算法计算每一个网页的 PageRank 值,反复进行迭代,直至结果收敛。显然,PageRank 值越大,该页面权威性越高。该算法与用户查询条件无关,只是给出每一页面的等级 PageRank 值,作为搜索引擎结果排序的一个参考,等级越高的页面排序越靠前。[7]

4 PageRank 算法的改进

PageRank 算法是独立于用户查询的,可以对用户要求产生快速的响应。不可避免的存在一些缺陷,如该算法偏重旧网页、完全忽略网页内容等,导致搜索结果精度不够,缺少有效信息的提供,不能很好的满足用户的搜索初衷。

时间权值:

针对 PageRank 算法偏重旧网页的问题，本文提出一个时间权值 T，T(j) 表示网页 j 的时间权值。考虑到人们一般对信息的新旧时间划分点，计算原理如下：

(1) T_{now} 为用户的即时查询时间（如：2008－1－1），T_{jb} 为网页 j 中的内容发布的时间。如果没有内容发布时间，则取网页 j 的修改时间；

(2) T_d 为时间间隔，$T_d = T_{now} - T_{jb}$。时间间隔以月为单位，忽略天数的差距，如 T_{now} 为 2008－1－2，T_{jb} 为 2007－9－28，则为 $T_d = (2008 - 2007) * 12 + (1 - 9) = 4$；

(3) 网页 j 的时间权值 T(j) 设置如下：

$$T(j) = \begin{cases} 1, & T_d \leqslant 12 \\ \dfrac{T_d}{12}, & T_d > 12 \end{cases}$$

4.1 内容权值

由于 PageRank 算法是完全忽略网页内容的，这就造成搜索结果的不理想，大部分的搜索结果不是用户想要的，或者一些比较靠近用户目的搜索结果排列在靠后的位置，要用户人为的从搜索结果中找出自己需要的信息是非常费时的，对用户来说造成了不便。对此，本文设计内容权值 W，它是根据文本内容与查询词的相关度来确定的，网页 j 的内容权值为 W(j)。一般来说一个网页的文本内容一般包括：标题、正文。一些论文网站的页面内容还包括：关键词、参考文献、摘要，即一个权值向量，这里运用本体论概念[8]。将这五个位置别设定权值，内容权值则根据查询词是否出现在这五个不同的位置来确定，下面对五个位置设定权值：

位置	权值
标题	0.8
正文	0.6lg(N+1)
关键词	0.3
参考文献	0.2
摘要	0.1

N 表示查询词在正文位置出现的个数。将标题和正文权值设定比较大而关键词、参考文献和摘要的权值设定比较小是为了使内容权值不失一般性。普通的网页都只有标题和正文两部分，关键词、参考文献、摘要都是比较正规的文章才有的特性，比较权威，这三个位置的权值可以对正规文章网页的内容权值起到一个比较小幅度的提高，不会对整个搜索结果的排序造成大的影响。

其计算原理：判断查询词是否出现在这五个位置中，若是，则增加相应得权值，否增加零。

例：如果查询词在网页 j 的标题和正文中出现，而且正文中的出现该查询词的个数为 8，因为该网页 j 的格式中不含关键词、参考文献、摘要，所以判断为查询词不出现在这三个位置。

$W(j) = 1*0.8 + 1*0.6*\lg(8+1) + 0*0.3 + 0*0.2 + 0*0.1 = 1.3724$

4.2 改进的 PageRank 算法

将一个网页的等级分成两个部分，一个部分是计算获得的 PR 值，另一是网页根据查询词得出的网页自评值，自评值 = W(j)/T(j)。从自评值中可以看出，与查询词关联度越大，内容权值就越大，自评值就越大；页面越旧，时间权值就越大，自评值就越小。

网页的最终等级 CTPR(j) = PR(j) * (W(j)/T(j))。

改进后的 PageRank 算法如下：

(1) 所有的网页初始值设为 1；

(2)

PR(A) = (1－d) + d((CTPR(T_1)/C(T_1)) + LCTPR(T_n)/C(T_n))CTPR(i) = PR(i) * (W(i)/T(i))

将 CTPR 值作为搜索引擎结果排序的一个参考，等级越高的页面排序越靠前。

5 实际效果分析

在 Google 搜索网站上进行“Web 结构挖掘”搜索，搜索结果如下：

从这个结果上看,这个排列顺序是不理想的,我对其中的两个网页进行分析,一个是"网络应用 – 远程教育",另一个是"计算机科学论坛 – Web 结构挖掘算法概述及应用"。即第一条和第五条。

分别对这两个网页进行 PR 值查询,查询网址为 http://www.thinkpage.cn/pagerank/。其结果如下:

第一个网页:

http://www.edu.cn/20060801/3202216.shtml 的 Google PageRank 值为 5;

第五个网页:

http://www.ieee.org.cn/dispbbs.asp?boardID=69&ID=48601 的 Google PageRank 值为 3;

现在使用 PageRank 改进算法 CTPR 进行计算,分析如下:第一个网页的建立时间是 2006 – 8 – 1,"Web 结构挖掘"出现在正文处,且数量为 4;第五个网页的建立时间是 2006 – 6 – 16,

"Web 结构挖掘"出现在标题,摘要,关键词,正文,正文中的个数为 5,有参考文献。

若用户查询时间为 2008 – 1 – 2,则第一个网页的时间距离为(2008 – 2006) * 12 + (1 – 8) = 17,时间权值为 T(1) = 17/12 = 1.416;第五个网页的时间距离为(2008 – 2006) * 12 + (1 – 6) = 19,时间权值为 T(5) = 19/12 = 1.583。

第一个网页的内容权值

W(1) = 0 * 0.8 + 0 * 0.6 * lg(4 + 1) + 0 * 0.3 + 0 * 0.2 + 0 * 0.1 = 0.42;

第五个网页的内容权值

W(5) = 1 * 0.8 + 1 * 0.6 * lg(5 + 1) + 1 * 0.3 + 1 * 0.2 + 1 * 0.1 = 1.86;

最终的 CTPR 值计算如下:

CTPR(1) = PR(1) * W(1)/T(1) = 5 * 0.42/1.416 = 1.483;

CTPR(5) = PR(5) * W(5)/T(5) = 3 * 1.86/1.583 = 3.525;

从以上得出的结果中可以看出,CTPR(5) > CTPR(1),认为第五个网页更加靠近用户查询的初衷,应该将第五个网页排在第一个网页的前面。事实上,人为地分析这两个网页,也认为第五个网页要优于第一个网页,说明 CTPR 算法是有效的。

6 总结

随着 Internet 的飞速发展,Web 挖掘成为当前的研究热点,而 Web 结构挖掘在 Web 挖掘中扮演了重要的角色。本文就对其中最常用的 PageRank 算法进行了改进,提高了已有算法的有效性。

改进的 PageRank 依然是针对 Web 页面的链接进行分析,增加了 Web 页面的内容的分析和网页修改时间的计算,肯定会增加搜索时间。从效果上看,它使搜索结果更加丰富,查询结果按链接、内容、时间差三者的综合结果排列,能让一些新的、比较权威的文本出现在靠前的位置,容易被用户发现,更加接近用户的查询初衷。改进的算法对于比较精确的查询词有较好的效果,对于模糊词方面没有明显的改进,需要进一步研究。

参考文献

1 麦小冬,余海冰. Web 数据挖掘综述. 科技咨询导报,2007(1):14 – 15.

2 陈学进. Web 结构挖掘研究. 安徽. 合肥工业大学,2006.

3 王艳华,张纪. Web 结构挖掘及其算法. 计算机工程,2005,(7):125 – 127.

4 Jason V. Davis, Inderjit S. Dhillon. Estimating the Global PageRank of Web Communities. ACM. KDD' 06, August 20 – 23, 2006,116 – 125.

5 Michael Brinkmeier. PageRank Revisited. ACM Transactions on Internet Technology,2006,6(3):282 – 301.

6 Ricardo Baeza – Yates. Paolo Boldi and Carlos Castillo. Generalizing PageRank: Damping Functions for Link – Based Ranking Algorithms. ACM. SIGIR' 06, August 6 – 11, 2006. 308 – 315.

7 吴春旭,郭磊. Web 结构挖掘的 PageRank 算法改进. 情报技术,2005,(10):55 – 58.

8 Soumen Chakrabarti. Dynamic Personalized Pagerank in EntityRelation Graphs. ACM. WWW 2007, May 8 – 12, 2007. 571 – 580.